

UNITED STATES PATENT APPLICATION

for

**AUTOMATICALLY CONFIGURING A COMPUTER FIREWALL
BASED ON NETWORK CONNECTION**

Applicants:

Mark J. McArdle
Brent A. Johnston
Philip D. R. Nathan
James Dool

prepared by:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN
12400 Wilshire Boulevard
Los Angeles, CA 90026-1026
(408) 720-8598

EXPRESS MAIL CERTIFICATE OF MAILING

"Express Mail" mailing label number EL627533101US

Date of Deposit MARCH 8, 2001

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231.

MICHELLE BEGAN
(Typed or printed name of person mailing paper or fee)

Michelle Began
(Signature of person mailing paper or fee)

AUTOMATICALLY CONFIGURING A COMPUTER FIREWALL BASED ON NETWORK CONNECTION

FIELD OF THE INVENTION

5 This invention relates generally to firewalls for networked computers, and more particularly to automatically configuring a firewall for a networked computer.

COPYRIGHT NOTICE/PERMISSION

10 A portion of the disclosure of this patent document contains material which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever. The following notice applies to the software and data as described below and in the drawings hereto: Copyright © 2000, Network Associates, Inc., All Rights
15 Reserved.

BACKGROUND OF THE INVENTION

20 When a computer is connected to a public network, such as the Internet, the computer is vulnerable to attack and compromise from other computers on the network. In contrast, if the same computer is connected to a private network, such as a corporate local area network (LAN), the security risks are much fewer, and features like file and printer sharing are common. Therefore, firewall software has been developed to protect the computer and such software is configurable by the user to determine the level of security desired.

In many cases, the computer may be connected to different networks at different times. For example, a notebook computer that is connected to a corporate LAN when the user is in the office may be connected to a public network when the user is working at home or traveling. The security configuration for the computer when it is used in the office would allow most all network traffic to and from the LAN because the LAN is a trusted network. On the other hand, when the computer is connected to a public network, the security configuration should be changed to prevent almost all unsolicited traffic because the public network is untrusted.

Using prior art firewall software, the user must manually re-configure the security settings when changing network connections, posing significant risks and challenges in protecting the computer. If the user attempts to connect to a trusted network with the firewall configured for an untrusted network, the user probably cannot log into the corporate LAN because it is likely that some of the protocols allowed on the LAN are prohibited on the untrusted network. More critically, if the firewall software is configured for a trusted network when the user connects the computer notebook to an untrusted network, the computer is now much less protected from attack.

SUMMARY OF THE INVENTION

A firewall protecting a computer is automatically configured to a particular security policy based on an address assigned to a network adapter. A zone that contains the network address is determined and a security policy that is assigned to the zone is associated with the network adapter. The security policy specifies the configuration the firewall uses when handling network traffic between the adapter and the network

corresponding to the address. In another aspect, the address assigned to the network adapter is also determined.

Because higher levels of protection can be specified in security policies for zones that encompass untrusted networks, the computer is protected from attacks from an untrusted network when it switches from a trusted one. Since the firewall is configured automatically, the computer is not blocked from accessing the trusted network when it switches back. Furthermore, the user no longer has to remember how the firewall was last configured and then manually change the security settings if necessary. Thus, the risks of switching connections from a trusted to an untrusted network are reduced and the complexities of firewall management are hidden from the user.

The present invention describes systems, clients, servers, methods, and computer-readable media of varying scope. In addition to the aspects and advantages of the present invention described in this summary, further aspects and advantages of the invention will become apparent by reference to the drawings and by reading the detailed description that follows.

BRIEF DESCRIPTION OF THE DRAWINGS

FIGs. 1A and 1B are diagrams illustrating a system-level overview of an embodiment of the invention;

FIGs. 2A and 2B are flowcharts of methods to be performed by a client computer according to an embodiment of the invention;

FIG. 3 is a diagram of a policy data structure for use in an implementation of the invention;

FIG. 4A is a diagram of one embodiment of an operating environment suitable for practicing the present invention; and

FIG. 4B is a diagram of one embodiment of a computer system suitable for use in the operating environment of FIG. 4A.

5

DETAILED DESCRIPTION OF THE INVENTION

In the following detailed description of embodiments of the invention, reference is made to the accompanying drawings in which like references indicate similar elements, and in which is shown by way of illustration specific embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that logical, mechanical, electrical, functional, and other changes may be made without departing from the scope of the present invention. The following detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims.

A system level overview of the operation of an embodiment of the invention is described by reference to FIGs. 1A and 1B, which illustrate a networked client computer 101 that connects to two different networks through a network adapter 105. One is a trusted network 111, such as a privately controlled corporate local-area network (LAN). The other is an untrusted network 113, such as the Internet, that exposes the computer 101 to security attacks.

Firewall software 103 in the computer 101 filters all network traffic passing between the computer 101 and a network in accordance with a security policy. When the network adapter 105 connects to a network, the firewall software 103 queries the network

adaptor 105 for the network address of the computer. If the network adaptor 105 responds with an address within a zone of addresses for the trusted network 111 (as illustrated by arrow 107 in FIG. 1A), the firewall automatically configures itself according to a security policy 109 designed for a trusted network 111. On the other hand, if the address returned
5 is within the zone of addresses for the untrusted network 113 (as illustrated by arrow 115 in FIG. 1B), the firewall automatically configures itself in accordance with a security policy 117 assigned to the untrusted network 113. Thus, the computer 101 is protected from attacks from the untrusted network 113 without the user having to manually reconfigure the firewall software 103 when switching connections from a trusted to an
10 untrusted network.

Although only two networks are illustrated in FIGs. 1A and 1B, a mixture of multiple trusted networks, multiple trusted, and/or untrusted security policies are contemplated and within the scope of the invention. Furthermore, a computer may have more than one network adapter installed and the invention handles each one independently
15 so that traffic addressed to each network adapter is filtered according to the security policy assigned to the particular adapter. It will be appreciated that the invention does not depend on the manner in which the network address is assigned to the adapter and is applicable to both static and dynamic assigned addresses.

Next, methods to be performed by the user, or client, computer in accordance with
20 one embodiment of the invention are described in terms of computer software with reference to flowcharts shown in FIGs. 2A and 2B. The methods constitute computer programs made up of computer-executable instructions. Describing the methods by reference to a flowchart enables one skilled in the art to develop such programs including such instructions to carry out the methods on suitably configured computers (the processor

of the computer executing the instructions from computer-readable media). The computer-executable instructions may be written in a computer programming language or may be embodied in firmware logic. If written in a programming language conforming to a recognized standard, such instructions can be executed on a variety of hardware platforms and for interface to a variety of operating systems. In addition, the present invention is not described with reference to any particular programming language. It will be appreciated that a variety of programming languages may be used to implement the teachings of the invention as described herein. Furthermore, it is common in the art to speak of software, in one form or another (e.g., program, procedure, process, application, module, logic...), as taking an action or causing a result. Such expressions are merely a shorthand way of saying that execution of the software by a computer causes the processor of the computer to perform an action or to produce a result. It will be further appreciated that the acts described in conjunction with FIGs. 2A and 2B are not required to be performed in the particular order shown and that the processes of the invention may be divided into more or fewer logical blocks than those shown.

The address zones for the networks may be defined by a systems administrator or by the user. An address zone can be defined positively or negatively. In one embodiment, an address zone is bounded through a range of addresses (or a network subnet). In another embodiment, an address zone is defined by a list of addresses. In an embodiment for TCP/IP and similar networks, all addresses within specified subnet having an address greater than or equal to a defined starting address are considered within a particular zone. Alternatively, the address zone for a network may be defined negatively by listing excluded addresses, with all other addresses being considered within the zone. A zone does not have to be explicitly defined since all addresses not within a defined zone are

considered to be within a default zone. In one embodiment, initially all network addresses are considered within a default zone and are considered part of an untrusted network.

Similarly, the security policies may be based on a default configuration or tailored to the requirements of the systems administrator or the user. One embodiment of a zone and policy administration method 200, illustrated in FIG. 2A, is invoked by the systems administrator or the user to create or change policies, zones, and the assignment of policies to zones. If input to the administration method 200 is policy information (block 201), the method 200 creates or changes the appropriate policy (block 203). If the input is zone information (block 205), the method 200 creates or changes the appropriate network zone (block 207). Otherwise, the input is assignment information, which causes a selected policy to be assigned to a selected zone (block 209). If there is no further processing to be performed (block 211), the method 200 ends.

In an alternate embodiment not shown, the policies, zones, and assignments can be “locked down” so that only authorized users can change them. In still another embodiment, the administration method 200 runs on a server computer and the system administrator defines the policies, zones, and assignments, which are subsequently transparently downloaded to the client computer as part of a standard automated update routine performed when the client is connected to a particular network location.

A firewall configuration method 220 illustrated in FIG. 2B is invoked to determine what security policy to apply to network traffic through the network adapters installed in the client computer. The method 220 reads a policy file containing the policies, zones, and assignments at block 201. An exemplary embodiment of a policy file is described further below. The method 220 loops through all the network adapters on the computer (block 223 through block 231), determining the network address of each (block 225) and the zone

to which the network address belongs (block 227). The method 220 associates the adapter with the security policy assigned to the zone (block 229).

Once the appropriate security policy has been associated with a network adapter, the firewall filters incoming and outgoing network traffic as specified by the security policy. One of skill in the art will immediately recognize that the method 220 may be invoked at various times, such as when the computer is booted, when a network connection is initially made, when the policy settings are updated, and/or when a network connection changes.

The processing represented block 225 depends on the operating system executing on the computer. For example, in a computer executing Microsoft Windows NT or Windows 2000, the network adapters installed on the computer and their current network addresses are stored in an operating system registry. Thus, in one embodiment for Windows NT/2000, the method 200 maps each adapter registry identifier to its network address in another section of the registry. The Windows 95 and Windows 98 registries do not have similar entries for installed network adapters. Therefore, in one Windows 9x embodiment, the processing at block 225 monitors network traffic and examines the initial network traffic for each adapter to determine the network address. Embodiments for other operating systems may invoke standard system calls at block 225 to determine the network address. In an alternate embodiment of method 220 not shown, the adapter device drivers would modified to invoke and send a network addresses to the method 220 when the corresponding adapter connects to a network.

Turning now to FIG. 3, one embodiment of a policy file data structure 300 is described that contains the information for policies, zones, and assignments. In one particular embodiment, the policy file data structure 300 is stored as an XML (extensible

markup language) document in the file system of the client computer although the invention is not so limited.

A security policy section 301 of the policy file data structure 300 contains an entry for each security policy. A security policy is identified by a policy identifier field 309 and is specific to a particular network protocol that is identified in a protocol identifier field 311. The security policy specifies filters for some or all of ports and/or services defined by the protocol (collectively referred to as protocol “elements”). Each relevant port 313 and service 315 within a security policy is represented by an element identifier field 303, a field containing the filter settings 305, and a log indicator field 307. For example, a security policy for a TCP/IP network would include the PPTP (point-to-point tunneling protocol), RIP (routing information protocol), DHCP (dynamic host configuration protocol), ARP (address resolution protocol), Ident (identification protocol), ICMP (internet control message protocol) and VPN (virtual private networking) ports, and the NetBIOS (network basic input/output system) service. The default settings for a “high” security policy on a TCP/IP network would disallow incoming network traffic through the PPTP and ICMP ports, allow incoming network traffic through the RIP, DHCP, ARP and VPN ports, and disallow access through NetBIOS to shared resources on the computer and disallow the computer from using the shared resources of other computer on the network. Additionally, incoming network traffic that attempted to access the computer using PPTP and NetBIOS would be logged. In addition to globally allowing or disallowing network traffic through a port for a particular protocol, finer granularity filters may be achieved by specifying one or more source or destination addresses in the filter settings 305. The resulting security policy is based on a combination of ports and network addresses, such as

“allow ICMP only to or from the specified addresses” or “disallow PPTP to or from the specified addresses.”

5 A zone section 321 of the policy file data structure 300 contains an entry for each defined address zone. A zone entry includes an identifier field 323, a field 325 for the address parameters that define the zone, and an identifier field 237 for the security policy assigned to the zone. As described above, a default zone may be defined by addresses that are not within any other zone. In this case, the address parameters field 325 for the default zone entry would be empty. Thus, if an Internet zone is defined and the high security policy described previously for a TCP/IP network is assigned to the Internet zone, the
10 invention would shield the computer from attacks using the high security policy when the computer was attached to the Internet.

The following description of FIGs. 4A-B is intended to provide an overview of computer hardware and other operating components suitable for implementing the invention, but is not intended to limit the applicable environments. One of skill in the art
15 will immediately appreciate that the invention can be practiced with other computer system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, and the like. The invention can also be practiced in distributed computing environments where tasks are performed by remote processing
20 devices that are linked through a communications network having a physical or wireless infrastructure, or a combination of both.

FIG. 4A shows several computer systems that are coupled together through a network 3, such as the Internet. The term "Internet" as used herein refers to a network of networks which uses certain protocols, such as the TCP/IP protocol, and possibly other

protocols such as the hypertext transfer protocol (HTTP) for hypertext markup language (HTML) documents that make up the World Wide Web (web). The physical connections of the Internet and the protocols and communication procedures of the Internet are well known to those of skill in the art. Access to the Internet 3 is typically provided by Internet service providers (ISP), such as the ISPs 5 and 7. Users on client systems, such as client computer systems 21, 25, 35, and 37 obtain access to the Internet through the Internet service providers, such as ISPs 5 and 7, through either physical or wireless interfaces. Access to the Internet allows users of the client computer systems to exchange information, receive and send e-mails, and view documents, such as documents which have been prepared in the HTML format. These documents are often provided by web servers, such as web server 9 which is considered to be "on" the Internet. Often these web servers are provided by the ISPs, such as ISP 5, although a computer system can be set up and connected to the Internet without that system being also an ISP as is well known in the art.

The web server 9 is typically at least one computer system which operates as a server computer system and is configured to operate with the protocols of the World Wide Web and is coupled to the Internet. Optionally, the web server 9 can be part of an ISP which provides access to the Internet for client systems. The web server 9 is shown coupled to the server computer system 11 which itself is coupled to web content 10, which can be considered a form of a media database. It will be appreciated that while two computer systems 9 and 11 are shown in FIG. 4A, the web server system 9 and the server computer system 11 can be one computer system having different software components providing the web server functionality and the server functionality provided by the server computer system 11 which will be described further below.

Client computer systems 21, 25, 35, and 37 can each, with the appropriate web browsing software, view HTML pages provided by the web server 9. The ISP 5 provides Internet connectivity to the client computer system 21 through the modem interface 23 which can be considered part of the client computer system 21. The client computer

5 system can be a personal computer system, a network computer, a Web TV system, a handheld wireless device, or other such computer system. Similarly, the ISP 7 provides Internet connectivity for client systems 25, 35, and 37, although as shown in FIG. 4A, the connections are not the same for these three computer systems. Client computer system 25 is coupled through a modem interface 27 while client computer systems 35 and 37 are part of a LAN. While FIG. 4A shows the interfaces 23 and 27 as generically as a "modem," it will be appreciated that each of these interfaces can be an analog modem, ISDN modem, cable modem, satellite transmission interface (e.g. "Direct PC"), radio frequency (RF), cellular, or other interfaces for coupling a computer system to other computer systems.

10 Client computer systems 35 and 37 are coupled to a LAN 33 through network interfaces 39 and 41, which can be Ethernet network or other network interfaces. The LAN 33 is also coupled to a gateway computer system 31 which can provide firewall and other Internet related services for the local area network. This gateway computer system 31 is coupled to the ISP 7 to provide Internet connectivity to the client computer systems 35 and 37. The gateway computer system 31 can be a conventional server computer system. Also, the

15

20 web server system 9 can be a conventional server computer system.

Alternatively, as well-known, a server computer system 43 can be directly coupled to the LAN 33 through a network interface 45 to provide files 47 and other services to the clients 35, 37, without the need to connect to the Internet through the gateway system 31.

FIG. 4B shows one example of a conventional computer system that can be used as a client computer system or a server computer system or as a web server system. It will also be appreciated that such a computer system can be used to perform many of the functions of an Internet service provider, such as ISP 5. The computer system 51

5 interfaces to external systems through the modem or network interface 53. It will be appreciated that the modem or network interface 53 can be considered to be part of the computer system 51. This interface 53 can be an analog modem, ISDN modem, cable modem, token ring interface, satellite transmission interface (e.g. "Direct PC"), radio frequency (RF), cellular, or other interfaces for coupling a computer system to other

10 computer systems. The computer system 51 includes a processing unit 55, which can be a conventional microprocessor such as an Intel Pentium microprocessor or Motorola Power PC microprocessor. Memory 59 is coupled to the processor 55 by a bus 57. Memory 59 can be dynamic random access memory (DRAM) and can also include static RAM (SRAM). The bus 57 couples the processor 55 to the memory 59 and also to non-volatile

15 storage x65 and to display controller x61 and to the input/output (I/O) controller x67. The display controller x61 controls in the conventional manner a display on a display device x63 which can be a cathode ray tube (CRT) or liquid crystal display. The input/output devices x69 can include a keyboard, disk drives, printers, a scanner, and other input and output devices, including a mouse or other pointing device. The display controller x61 and

20 the I/O controller x67 can be implemented with conventional well known technology. A digital image input device x61 can be a digital camera which is coupled to an I/O controller x67 in order to allow images from the digital camera to be input into the computer system 51. The non-volatile storage x65 is often a magnetic hard disk, an optical disk, or another form of storage for large amounts of data. Some of this data is

often written, by a direct memory access process, into memory 59 during execution of software in the computer system 51. One of skill in the art will immediately recognize that the term "computer-readable medium" includes any type of storage device that is accessible by the processor 55 and also encompasses a carrier wave that encodes a data
5 signal.

It will be appreciated that the computer system 51 is one example of many possible computer systems which have different architectures. For example, personal computers based on an Intel microprocessor often have multiple buses, one of which can be an input/output (I/O) bus for the peripherals and one that directly connects the processor 55
10 and the memory 59 (often referred to as a memory bus). The buses are connected together through bridge components that perform any necessary translation due to differing bus protocols.

Network computers are another type of computer system that can be used with the present invention. Network computers do not usually include a hard disk or other mass
15 storage, and the executable programs are loaded from a network connection into the memory 59 for execution by the processor 55. A Web TV system, which is known in the art, is also considered to be a computer system according to the present invention, but it may lack some of the features shown in FIG. 4B, such as certain input or output devices. A typical computer system will usually include at least a processor, memory, and a bus
20 coupling the memory to the processor.

It will also be appreciated that the computer system 51 is controlled by operating system software which includes a file management system, such as a disk operating system, which is part of the operating system software. One example of an operating system software with its associated file management system software is the family of

operating systems known as Windows[®] from Microsoft Corporation of Redmond,
Washington, and their associated file management systems. The file management system
is typically stored in the non-volatile storage 65 and causes the processor 55 to execute
the various acts required by the operating system to input and output data and to store data
5 in memory, including storing files on the non-volatile storage 65.

Automatic configuration of a computer firewall based on a address for a network
adapter has been described. Although specific embodiments have been illustrated and
described herein, it will be appreciated by those of ordinary skill in the art that any
arrangement which is calculated to achieve the same purpose may be substituted for the
10 specific embodiments shown. This application is intended to cover any adaptations or
variations of the present invention.

For example, those of ordinary skill in the art will appreciate that while the
invention as been described as configuring firewall software executing on the processing
unit of the computer the firewall is protecting, the present invention is equally applicable
15 to firewall software executing on a processing unit for a hardware firewall. Furthermore,
one of ordinary skill in the art will immediately recognize that “network adapter” is a
generic term for all network interfaces that connect a computer to any type of network
environment. Therefore, it is manifestly intended that this invention be limited only by the
following claims and equivalents thereof.